

AMENDMENTS TO THE CLAIMS

Please amend claims 1, 5, 9, 16, 23 and 27, as set forth below.

1. (Currently Amended) A system that allows a table and a materialized view to be available while the materialized view is being refreshed, the system comprising:
 - a materialized view that is derived at least in part from a table;
 - a refresh log that contains a plurality of entries, each of the plurality of entries corresponding to a change in the table, each of the plurality of entries comprising an epoch identifier adapted to synchronize the refresh log between refreshing operations; and
 - a refresh manager that performs a refresh operation on the materialized view in multiple steps by (a) successively reading a first subset of the plurality of entries indicated by a specific epoch identifier from the refresh log, (b) identifying a second subset of the plurality of entries from within the first subset of the plurality of entries, the second subset of the plurality of entries falling within a primary key value boundary and (c) applying the second subset of the plurality of entries to the materialized view.
2. (Previously Amended) The system set forth in claim 1, wherein the epoch identifiers comprise epoch numbers that have been created since a previous refresh operation on the materialized view.
3. (Original) The system set forth in claim 1, wherein the second subset of the plurality of entries is applied to the materialized view in a primary key order.
4. (Original) The system set forth in claim 1, wherein the refresh manager is adapted to distinguish between entries of the second subset of the plurality of entries that

have already been applied to the materialized view in previous transactions and entries of the second subset of the plurality of entries that have not been applied to the materialized view in the event of a failure of the refresh operation.

5. (Currently Amended) A method of refreshing a materialized view that is in part derived from a table, the method being adapted to improve the availability of the table and the materialized view while the materialized view is being refreshed, the method comprising:

deriving a materialized view from at least one table;

assigning an epoch identifier to changes made to the at least one table;

storing an entry corresponding to each change to the at least one table in a refresh log that includes a plurality of entries, each of the plurality of entries comprising an epoch identifier that is adapted to synchronize the refresh log between refreshing operations; and

performing a refresh operation in multiple operations, each of the multiple operations comprising (a) successively reading a first subset of the plurality of entries indicated by a specific epoch identifier from the refresh log, (b) identifying a second subset of the plurality of entries from within the first subset of the plurality of entries, the second subset of the plurality of entries falling within a primary key value boundary and (c) applying the second subset of the plurality of entries to the materialized view.

6. (Original) The method set forth in claim 5, comprising applying the second subset of the plurality of entries to the materialized view in a primary key order.

7. (Original) The method set forth in claim 5, comprising defining the epoch identifier to correspond to changes that have been made to the table since a previous refresh operation on the materialized view.

8. (Original) The method set forth in claim 5, comprising distinguishing between entries of the second subset of the plurality of entries that have already been applied to the materialized view in previous transactions and entries of the second subset of the plurality of entries that have not been applied to the materialized view in the event of a failure of the refresh operation.

9. (Currently Amended) A system that provides availability of a table and a materialized view while the materialized view is being refreshed, the table being derived at least in part from the materialized view, the system comprising:

a refresh log that contains a plurality of entries, wherein the plurality of entries comprise data that is being refreshed, each of the plurality of entries comprising an epoch identifier adapted to synchronize the refresh log between refreshing operations; and

a refresh manager that computes a table delta based on the refresh log and applies the table delta to the materialized view.

10. (Original) The system set forth in claim 9, wherein each of the plurality of entries comprises an epoch identifier.

11. (Original) The system set forth in claim 10, wherein the epoch identifier corresponds to changes that have been made to the table since a previous refresh operation on the materialized view.

12. (Original) The system set forth in claim 9, wherein the table delta is applied to the materialized view in a primary key order.

13. (Original) The system set forth in claim 9, wherein the table delta is used to refresh the materialized view in multiple transactions.

14. (Original) The system set forth in claim 9, wherein a primary key value for each entry from the refresh log is recorded after that entry is applied to the materialized view.

15. (Original) The system for refreshing the materialized view set forth in claim 9, wherein the refresh manager is adapted to distinguish between a first subset of the plurality of entries that have already been applied to the materialized view in previous transactions and a second subset of the plurality of entries that have not been applied to the materialized view in the event of a failure of the refresh operation.

16. (Currently Amended) A method of refreshing a materialized view that is derived at least in part from a table, the method being adapted to provide availability of the table and the materialized view while the materialized view is being refreshed, the method comprising the acts of:

storing a plurality of entries corresponding to changes in the table in a refresh log, wherein the plurality of entries comprise data that is being refreshed, each of the plurality of entries comprising an epoch identifier adapted to synchronize the refresh log between refreshing operations;

computing a table delta based on the refresh log;

refreshing the materialized view based on the table delta.

17. (Original) The method set forth in claim 16, wherein the table delta is applied to the materialized view in a primary key order.

18. (Original) The method set forth in claim 16, comprising updating the materialized view in multiple transactions.

19. (Original) The method set forth in claim 16, comprising storing an epoch identifier as a portion of each of the plurality of entries.

20. (Original) The method set forth in claim 19, comprising defining the epoch identifier to correspond to changes that have been made to the table since a previous refresh operation on the materialized view.

21. (Original) The method set forth in claim 16, comprising recording the primary key value for each entry from the update log after that entry is applied to the materialized view.

22. (Original) The method set forth in claim 16, comprising distinguishing between a first subset of the plurality of entries that have already been applied to the materialized view in previous transactions and a second subset of the plurality of entries that have not been applied to the materialized view in the event of a failure of the act of refreshing the materialized view.

23. (Currently Amended) A system that provides availability of a table and a materialized view while the materialized view is being refreshed, the table being derived at least in part from the materialized view, the system comprising:

a refresh log that contains a plurality of entries, wherein the plurality of entries comprise data that is being refreshed, each of the plurality of entries comprising an epoch identifier adapted to synchronize the refresh log between refreshing operations; and
means for computing a table delta based on the refresh log; and
means for applying the contents of the table delta to the materialized view.

24. (Original) The system set forth in claim 23, wherein each of the plurality of entries comprises an epoch identifier.

25. (Original) The system set forth in claim 24, wherein the epoch identifier corresponds to changes that have been made to the table since a previous refresh operation on the materialized view.

26. (Original) The system set forth in claim 23, wherein the means for applying the table delta to the materialized view is adapted to distinguish between a first subset of the plurality of entries that have already been applied to the materialized view in previous transactions and a second subset of the plurality of entries that have not been applied to the materialized view in the event of a failure of applying the table delta to the materialized view.

27. (Currently Amended) A computer readable medium, comprising:
a refresh log stored on the computer readable medium, the refresh log containing a plurality of entries, each of the plurality of entries comprising an epoch identifier adapted to

synchronize the refresh log between refreshing operations, wherein one of the plurality of entries comprises refreshable data associated with a materialized view; and
code adapted to refresh the materialized view at least in part from a table by computing a table delta based on the refresh log and applying the table delta to the materialized view.

28. (Original) The computer program set forth in claim 27, wherein each of the plurality of entries comprises an epoch identifier.

29. (Original) The computer program set forth in claim 28, wherein the epoch identifier corresponds to changes that have been made to the table since a previous refresh operation on the materialized view.

30. (Original) The computer program set forth in claim 27, wherein the refresh manager is adapted to distinguish between a first subset of the plurality of entries that have already been applied to the materialized view in previous transactions and a second subset of the plurality of entries that have not been applied to the materialized view in the event of a failure of a refresh operation.